

Tree Codes Improve Convergence Rate of Consensus Over Erasure Channels

Ravi Teja Sukhavasi and Babak Hassibi
California Institute of Technology

Abstract

We study the problem of achieving average consensus between a group of agents over a network with erasure links. In the context of consensus problems, the unreliability of communication links between nodes has been traditionally modeled by allowing the underlying graph to vary with time. In other words, depending on the realization of the link erasures, the underlying graph at each time instant is assumed to be a subgraph of the original graph. Implicit in this model is the assumption that the erasures are symmetric: if at time t the packet from node i to node j is dropped, the same is true for the packet transmitted from node j to node i . However, in practical wireless communication systems this assumption is unreasonable and, due to the lack of symmetry, standard averaging protocols cannot guarantee that the network will reach consensus to the true average. In this paper we explore the use of channel coding to improve the performance of consensus algorithms. For symmetric erasures, we show that, for certain ranges of the system parameters, repetition codes can speed up the convergence rate. For asymmetric erasures we show that tree codes (which have recently been designed for erasure channels) can be used to simulate the performance of the original “un erased” graph. Thus, unlike conventional consensus methods, we can guarantee convergence to the average in the asymmetric case. The price is a slowdown in the convergence rate, relative to the unerased network, which is still often faster than the convergence rate of conventional consensus algorithms over noisy links.

I. INTRODUCTION

In a network of agents, consensus refers to the process of achieving agreement between the agents in a distributed manner. Consensus problems, and in particular the problem of reaching consensus on the average of the values of the agents, have been around for a while and are often used to serve as a test case for studying distributed computation and decision making between a group of nodes/processors/dynamical systems ([1]–[6]). Most of the work in this area assumes that the agents are connected via a fixed underlying graph or network. In many applications, however, the links in the underlying graph are noisy or unreliable. In the context of consensus problems, the unreliability of communication links between nodes has been traditionally modeled by allowing the underlying graph to vary with time. In other words, at each time instant some of the links are allowed to be erased, and depending on the realization of the

link erasures, the underlying graph at each time instant is assumed to be a subgraph of the original graph. Furthermore, the distributed algorithm for reaching consensus remains unchanged: the same distributed averaging algorithm is used, except that only the information received at each time is used. An important assumption that is implicitly made in this model is that the erasures are symmetric: if at time t the packet from node i to node j is dropped, the same is true for the packet transmitted from node j to node i . In practical wireless communication systems this assumption is patently unreasonable: the additive noise at the two nodes are independent and, furthermore, communication in the two directions occurs at either different times or over different frequency bands. If standard averaging protocols are performed, this loss of symmetry can prohibit the network from reaching consensus to the true average (standard consensus protocols require that the “update” matrix be doubly stochastic, something that cannot be guaranteed in the asymmetric case).

The goal of this paper is to explore the use of channel coding to improve the performance of consensus algorithms, especially in the asymmetric case. A major impetus for this work is the recently designed tree codes for erasure channels [7], which, as we demonstrate, resolves the problem encountered in the asymmetric case.

For asymmetric erasures we show that tree codes can be used to simulate the performance of the original *unerased* graph. Thus, unlike conventional consensus methods, we can *guarantee* convergence to the average in the asymmetric case. As expected, the price is a slowdown in the convergence rate, relative to the convergence rate of the unerased network. Nonetheless, the slowdown is still often faster than the convergence rate of conventional consensus algorithms over erasure links.

II. PROBLEM SETUP

Consider a group of N nodes denoted by $\mathcal{N} = \{1, 2, \dots, N\}$. We assume that the nodes are connected by an undirected communication graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ which is often referred to as the interaction graph. Throughout the analysis \mathcal{G} is assumed to be time invariant. Let $A = [a_{ij}]$ denote the adjacency matrix of G , i.e., $a_{ij} = 1$ if $(i, j) \in \mathcal{E}$ and 0 otherwise. Note that $a_{ii} = 0$. Let x_0^i denote the initial value at node i . The objective is for the nodes to compute the global average $r = \frac{1}{N} \mathbf{1}^T x_0$, where $\mathbf{1}$ denotes an N -dimensional column of ones and x_0 is the column vector of the x_0^i 's. We model the communication links between nodes as packet erasure links. Further, we ignore quantization effects due to packetization. The standard packet sizes in practice justify this assumption. We denote the event of successful packet reception from node j to node i at time k with the Bernoulli random variable X_k^{ij} , i.e., $X_k^{ij} = 1$ if the packet is received successfully at time k and 0 otherwise. This notation is summarized in Table I.

TABLE I
NOTATION

$\ y\ , y \in \mathbb{R}^N$	$\sqrt{\sum_{i=1}^N y_i^2}$, i.e., the two norm of y
$\mathcal{N} = \{1, 2, \dots, N\}$	the set of nodes
$\mathcal{G} = (\mathcal{N}, \mathcal{E})$	the underlying communication graph
$A = [a_{ij}]$	the adjacency matrix of G , i.e., $a_{ij} = 1$ if $(i, j) \in \mathcal{E}$ and 0 otherwise
Δ	largest degree of any vertex in G
x_0^i	the initial value at node i
x_0	column of x_0^i 's
r	the initial average, i.e., $\frac{1}{N} \mathbf{1}^T x_0$
p	packet erasure probability
X_k^{ij}	1 if the packet sent from node j to node i at time k is successfully received and 0 o.w
$\rho(\cdot)$	spectral radius of a matrix
$A \circ B$	Hadamard product, i.e., $(A \circ B)_{ij} = A_{ij} B_{ij}$
$A \otimes B$	Kronecker product
$D(p, q)$	$p \log \frac{p}{q} + (1-p) \log \frac{1-p}{1-q}$ i.e., Kullbeck Leibler divergence between Bernoulli(p) and Bernoulli(q)

III. BACKGROUND

For a fixed communication graph \mathcal{G} , a typical algorithm to achieve consensus is of the following form.

$$x_{k+1}^i = w_{ii}x_k^i + \sum_j w_{ij}x_k^j \quad (1)$$

W obeys the underlying graph, i.e., for $i \neq j$, $W_{ij} = 0$ if $(i, j) \notin \mathcal{E}$. In other words, each node updates its value by taking a weighted sum of its own previous value with those of its neighbors. In short, the equation can be written as

$$x_{k+1} = Wx_k \quad (2)$$

Such an algorithm is said to achieve consensus if

$$\lim_{k \rightarrow \infty} x_k^i = r \triangleq \frac{1}{N} \sum_j x_0^j \quad (3)$$

In such a static setup where the weights and the underlying interaction graph does not change with time, it is well known that consensus is achieved if and only if

$$\lim_{k \rightarrow \infty} W^k = \frac{1}{N} \mathbf{1} \mathbf{1}^T \quad (4)$$

Further (4) holds if and only if the following conditions hold (e.g., see [8])

1) W is doubly stochastic, i.e.,

$$\mathbf{1}^T W = \mathbf{1}^T, \quad W \mathbf{1} = \mathbf{1} \quad (5)$$

2) $\rho(W - \frac{1}{N} \mathbf{1} \mathbf{1}^T) < 1$

Note that $x_k = W^k x_0$. Under the above conditions, $x_k \rightarrow \frac{1}{N} \mathbf{1} \mathbf{1}^T x_0 = r \mathbf{1}$. The convergence rate, $\mu(W)$, of the above consensus algorithm is formally defined as

$$\mu(W) = \sup_{x_0 \neq r \mathbf{1}} \lim_{k \rightarrow \infty} \left[\frac{\|x_k - r \mathbf{1}\|}{\|x_0 - r \mathbf{1}\|} \right]^{\frac{1}{k}} \quad (6)$$

and is given by $\mu(W) = \rho(W - \frac{1}{N} \mathbf{1} \mathbf{1}^T)$. There is a considerable amount of work that explores different choices of W and how it affects the rate of convergence of the consensus algorithm (e.g., [8]).

For the purpose of this paper and for ease of exposition, we use a specific but natural choice of W (e.g., [1]) given by $W = I - \epsilon \mathcal{L}$, where \mathcal{L} is the Laplacian of the interaction graph \mathcal{G} , i.e., $\mathcal{L} = D - A$. $D = \text{diag}\{\Delta_i\}$ where Δ_i is the degree of node i . Let $0 = \lambda_N(\mathcal{L}) \leq \lambda_{N-1}(\mathcal{L}) \leq \dots \leq \lambda_1(\mathcal{L})$ denote the eigen values of \mathcal{L} . The multiplicity of the zero eigen value is the number of connected components in the graph and $\lambda_{N-1}(\mathcal{L}) > 0$ if and only if the graph is connected.

For such a choice of W , the spectral radius is given by $\rho(W - \frac{1}{N} \mathbf{1} \mathbf{1}^T) = \max\{1 - \epsilon \lambda_{N-1}(\mathcal{L}), \epsilon \lambda_1(\mathcal{L}) - 1\}$. We state this as a Lemma for later reference.

Lemma 3.1: The convergence rate, μ , of (1) with $W = I - \epsilon \mathcal{L}$ is

$$\mu = \max\{1 - \epsilon \lambda_{N-1}(\mathcal{L}), \epsilon \lambda_1(\mathcal{L}) - 1\} \quad (7)$$

So, the conditions 1) and 2) above are satisfied if and only if $\epsilon < \frac{2}{\lambda_1(\mathcal{L})}$. Furthermore, the convergence rate μ is maximized when the two quantities in (7) coincide, i.e., when

$$\epsilon = \epsilon^* = \frac{2}{\lambda_1(\mathcal{L}) + \lambda_{N-1}(\mathcal{L})} \quad (8)$$

In particular, any $\epsilon < 1/\Delta$ will work where $\Delta = \max_i \Delta_i$. We remark that the techniques presented in the paper are independent of the choice of the weight matrix W . Whenever we wish to write closed form expressions for the

convergence rates, we use the specific choice $W = I - \epsilon^* \mathcal{L}$ for simplicity.

IV. COMMUNICATION MODEL

In practice, the communication links between nodes can be unreliable. Conventionally, this has been taken into account by allowing the interaction topology to change with time. So, at time k , the connectivity between nodes is described by the graph \mathcal{G}_k where \mathcal{G}_k can now vary with time. There is a considerable amount of literature on the problem of achieving consensus under such time varying interaction topologies ([2], [6], [9]–[11]). We model unreliable communication as packet erasures. So, at each time k , the packet transmitted from node i to, say, node j is either received ($X_k^{ji} = 1$) or erased ($X_k^{ji} = 0$). Similarly, the packet sent from node j to node i is either received ($X_k^{ij} = 1$) or erased ($X_k^{ij} = 0$). We consider two erasure models

- 1) *Symmetric*: $X_k^{ij} = X_k^{ji}$, and $X_k^{ij}, X_k^{m\ell}$ are independent of each other whenever $(i, j) \notin \{(m, \ell), (\ell, m)\}$
- 2) *Asymmetric*: $X_k^{ij}, X_k^{m\ell}$ are independent of each other whenever $(i, j) \neq (m, \ell)$, in particular X_k^{ij} and X_k^{ji} are independent.

The literature on consensus over time varying topologies only captures the symmetric case. Even though, consensus under very general conditions has been established, not much appears to be available by way of the rate of convergence. Under the asymmetric erasure model, the resulting interaction graph is effectively directed. An edge between node i and j is replaced by a pair of directed edges. The effective graph at any time depends on the packets that were erased in that round. Under this setup, we define the adjacency matrix $A = [a_{ij}]$ and the Laplacian \mathcal{L} as follows; $a_{ij} = 1$ if $(i \leftarrow j) \in \mathcal{E}$ and $\mathcal{L} = D - A$ with $D = \text{diag}\{\Delta_i\}$ and $\Delta_i = \sum_j a_{ij}$. The resulting adjacency matrix and the Laplacian are not symmetric in general. As a result, they are not doubly stochastic either, i.e., $\mathbf{1}^T \mathcal{L} \neq \mathbf{1}^T$. When the graph \mathcal{G} is directed, (Olfati-Saber Murray 2007) prove that average consensus is achieved using a fixed $W = I - \epsilon \mathcal{L}$ if and only if the interaction graph \mathcal{G} is balanced, i.e., the in-degree of each node is equal to its out degree (cite Olfati-Saber Murray 2007). But when the link failures are random, the resulting interaction graph will generally not be balanced at every time step. But with coding, one can overcome this problem as we will show later.

V. DOES CODING HELP?

It turns out coding does help. In fact, to study the effect of coding we need to distinguish between the symmetric and asymmetric erasure models. When the erasures are symmetric, i.e., when $X_k^{ij} = X_k^{ji}$, this means that node i (respectively, node j) *knows* what node j (respectively, i) has received. For example, if node i successfully received a packet from node j , it knows that node j also successfully received the packet intended for it; alternately if node i receives an erasure from node j , it knows that the packet intended for node j was also erased. In this case, the links between the different nodes are erasure links with feedback (where the transmitter knows what the receiver receives).

For erasure links with feedback it is well known that the optimal coding scheme is *retransmission*, i.e., the transmitter retransmits its packet until it is received at the receiver.

When the erasures are not symmetric, one needs a more sophisticated coding scheme (called tree codes). We shall further explain this below.

When there are erasures and when there is no coding, an iteration of the consensus algorithm at node i is given by

$$x_{k+1}^i = x_k^i - \epsilon \sum_j a_{ij} X_k^{ij} (x_k^i - x_k^j) \quad (9)$$

The effective adjacency matrix at time k is then $A_k = A \circ X_k$, where $X_k = [X_k^{ij}]$. The associated Laplacian is $\mathcal{L}_k = D_k - A_k$ where $D_k^i = \sum_j A_k^{ij} = \sum_j a_{ij} X_k^{ij}$.

1) *Symmetric Erasures*: In this case, note that even without coding, the nodes achieve average consensus albeit at a slower rate depending on the erasure probability, say p . We show that coding (in this case retransmitting until successful reception) results in faster convergence whenever there exists a constant $R' > 0$ such that

$$D(1 - R', p) > \log(\Delta + 1) \quad (10)$$

$$\mu^{R'} < \sqrt{\lambda_2(\Gamma)} \quad (11)$$

where μ is as in (7), $H(\cdot)$ is the binary entropy function and Γ is defined in Lemma 8.1.

2) *Asymmetric Erasures*: Since X_k^{ij} and X_k^{ji} are independent, they are not equal in general. Note that $\mathcal{L}_k \mathbf{1} = \mathbf{1}$ but $\mathbf{1}^T \mathcal{L}_k \neq \mathbf{1}^T$ in general which violates (5). Furthermore, the associated graph is not balanced either, $\sum_j a_{ij} X_k^{ij} \neq \sum_i a_{ji} X_k^{ji}$, in general. In this case, the nodes will not achieve *average* consensus. But under very mild conditions, it is well known that the nodes achieve an agreement, i.e., $x_k \rightarrow Y \mathbf{1}$ where Y is a random variable that does not necessarily concentrate around the initial average r . But tree codes allow us to simulate the original recursions, i.e., (1), and hence guarantee asymptotic average consensus. Before proceeding further, we provide a brief introduction to tree codes.

VI. BACKGROUND ON TREE CODES

The problem of achieving consensus over erasure channels is an instance of the problem of simulating interactive communication protocols between a network of agents over unreliable links. In the specific case of consensus, the interactive communication protocol amounts to executing (1) at every node. In this context, Rajagopalan et al in [12] use *tree codes* to simulate such protocols with exponentially vanishing probability of error in the length of the protocol (e.g., the length of the protocol is said to be m if one needs to execute m iterations of (1)). Another very important instance of such interactive communication problems is one of stabilizing unstable dynamical systems over

noisy communication channels (cite Sahai here). Even though the central role of tree codes in such problems has been identified, there have been no practical constructions until very recently. In [7], [13], the authors proposed an explicit ensemble of *linear* tree codes with efficient decoding for the erasure channel. Equipped with this construction of tree codes, we can examine more closely how they can be used for specific problems such as consensus over erasure links which is what we do here. Before proceeding further, we will digress a little bit to outline the codes proposed in [13] and list their relevant properties.

A. Linear time-invariant tree codes

A tree code is essentially a semi-infinite causal encoding scheme which has a certain ‘Hamming distance’-like property. When decoding using maximum likelihood decoding over a discrete memoryless channel (DMC), such a tree code guarantees exponentially small error probability with delay. In other words, the probability of incorrectly decoding a symbol (or packet) d time steps in the past decays exponentially in d . If the rate of the code is $R < 1$, such a causal encoding/decoding scheme with such an exponentially decaying probability of error (exponent β say) is said to be (R, β) -anytime reliable. We will make this more precise below. We will describe the tree codes of (our work) in terms of their anytime reliability rather than in terms of their distance properties, because ultimately it is the exponent and rate that matter when communicating over DMCs. Since communication is packetized, let Λ denote the packet length. Each packet can be viewed as a symbol from \mathbf{F}_2^Λ . Suppose information is generated at the rate of nR packets per time instant at the encoder. Then a rate R time-invariant causal linear code is given by

$$c_t = G_1 b_t + G_2 b_{t-1} + \dots + G_t b_1, \quad t \geq 1 \quad (12)$$

where $c_t \in \mathbf{F}_2^{n\Lambda}$, $b_i \in \mathbf{F}_2^{nR\Lambda}$ and $G_i \in \mathbf{F}_2^{n\Lambda \times nR\Lambda}$. So, at each time, the encoder receives nR packets and transmits n packets. Note that this is essentially a convolutional code with infinite memory. The decoder, at each time t , generates estimates $\hat{b}_{\tau|t}$ for $1 \leq \tau \leq t$ where $\hat{b}_{\tau|t}$ denotes the decoder’s estimate of b_τ using the channel outputs received till time t .

Definition 1 (Anytime Reliability): A causal code as in (12) is said to be (R, β) -anytime reliable if

$$P\left(\hat{b}_{\tau|t} \neq b_\tau\right) \leq 2^{-\beta(t-\tau+1)}, \quad \forall \tau, t \geq d_o \quad (13)$$

for some fixed d_o independent of τ, t .

Let $p' = p^{1/\Lambda}$. In [13], the authors showed that if the entries of G_i are drawn i.i.d Bernoulli (1/2), then almost every code in this ensemble is (R, β) -anytime reliable for $R < 1 - p'$ and $\beta < n\Lambda E(R)$, where $E(R)$ is an exponent that depends on the DMC and that can be explicitly computed. For the packet erasure channel with erasure probability p ,

$E(R)$ is given by (see [13])

$$E(R) = \begin{cases} H^{-1}(1-R) \log \frac{1}{p'}, & R \leq \gamma_1 \\ 1 - \log(1+p') - R, & \gamma_1 \leq R \leq \gamma_2 \\ R \log \frac{R}{1-p'} + (1-R) \log \frac{1-R}{p'}, & \gamma_2 \leq R \leq 1-p' \end{cases} \quad (14)$$

where

$$\gamma_1 = 1 - H\left(\frac{p'}{1+p'}\right), \quad \gamma_2 = \frac{1-p'}{1+p'} \quad (15)$$

For the rest of the analysis, we will assume that we are given an (R, β) -anytime reliable code with $d_o = 0$.

VII. MAIN RESULTS

We present the results separately for the case of symmetric and asymmetric erasures.

A. Symmetric Link Failures

Note that the underlying interaction graph \mathcal{G} is fixed while each link is modeled as a packet erasure channel. The graph \mathcal{G} is assumed to be connected and the links are undirected. If all agents know that link failures are symmetric, then each link is effectively a packet erasure channel with feedback. In each communication round, node i would know that its packet transmission to node j is erased if it receives an erasure from node j in the same round. Recall that the consensus algorithm in the case where there are no erasures is given by

$$x_{k+1} = (I - \epsilon \mathcal{L})x_k \quad (16)$$

In particular, node i performs the algorithm

$$x_{k+1}^i = x_k^i - \epsilon \sum_j a_{ij}(x_k^i - x_k^j) \quad (17)$$

We now define the communication protocol.

1) *The Protocol:* A communication round is defined as one in which every node in the graph transmits one packet to each of its neighbors. The nodes are said to have completed m iterations if all of them successfully computed m iterations of (17). Note that this will in general take more than m communication rounds. Since each link is effectively an erasure channel with feedback, the optimal communication scheme at each node is to retransmit until successful reception. We describe this more precisely as follows. Let e denote an erasure. For each edge $j \rightarrow i$, we associate an input queue, Q_{in}^{ij} , and an output queue, Q_{out}^{ij} . $Q_{in,t}^{ij}$ contains the packets transmitted by node j to node i up to and including communication round t while $Q_{out,t}^{ij}$ contains the packets received by node i from node j .

$Q_{out,5}^{i,1} :$	x_0^1	x_1^1	e	e	x_2^1
$Q_{out,5}^{i,2} :$	x_0^2	e	x_1^2	w	x_2^2
$Q_{in,5}^{1,i} :$	x_0^i	x_1^i	w	x_2^i	x_2^i
$Q_{in,5}^{2,i} :$	x_0^i	x_1^i	x_1^i	x_2^i	w
past ← — — — → present					

Fig. 1. Consider an instance of the queues at node i . Suppose its only neighbors are nodes 1 and 2. In round 2, node i receives an erasure from node 2 and infers that its own transmission to node 2 must also have been erased. As a result, node i re-transmits x_1^i to node 2 in round 3. Similarly in round 3, node i knows that its transmission to node 1 was erased. Since the erased symbol was only a ‘wait’, node i does not re-transmit it in round 4. Instead, it checks if it can perform another iteration of (17). In this case, it can and hence transmits the new data x_2^i to node 1. In round 5, node i does not have any new data to transmit to node 2 and hence transmits a ‘wait’.

Also let b_t^{ij} denote the packet transmitted by node j to node i in communication round t and let z_t^{ij} denote the received packet. Then

$$z_t^{ij} = \begin{cases} b_t^{ij} & \text{w.p } 1-p \\ e & \text{w.p } p \end{cases} \quad (18)$$

Now if $z_t^{ji} = e$, then node j infers that b_t^{ij} was erased and hence retransmits it in the next communication round unless b_t^{ij} was a ‘wait’ symbol which we describe as follows. We say that a node i has ‘new data’ if it could compute one or more new iterations of (17). During communication rounds where node j does not have any new data to transmit, it transmits a wait symbol which we denote with w . The transmission from node i to node j in round t is described in Algorithm 1. Let \mathcal{N}_i denote the neighbors of node i , i.e., $\mathcal{N}_i = \{j' | a_{ij'} = 1\}$.

Algorithm 1 Node i ’s transmission to node j in round t

```

1: if  $z_{t-1}^{ji} = e$  and  $b_{t-1}^{ji} \neq w$  then
2:    $b_t^{ji} = b_{t-1}^{ji}$ , i.e., re-transmit
3: else
4:   For each  $j' \in \mathcal{N}_i$ , let  $\ell_{t,j'} = \max\{\ell' \mid x_{\ell'}^{j'} \in Q_{out,t}^{ij'}\}$ 
5:   Compute  $\ell_t = \min_{j' \in \mathcal{N}_i} \ell_{t,j'}$ 
6:   if  $\ell_t = \ell_{t-1} + 1$  then
7:     Compute  $x_{\ell_t+1}^i$  using (17) and set  $b_t^{ji} = x_{\ell_t+1}^i$  (note that  $\ell_t \leq \ell_{t-1} + 1$ )
8:   else
9:     i.e.,  $\ell_t = \ell_{t-1}$ , set  $b_t^{ji} = w$ 
10:  end if
11: end if

```

The algorithm is illustrated through an example in Fig 1. Using such an algorithm, we have the following bounds on the convergence rate of average consensus.

Theorem 7.1: Let $P_{M,R'}$ denote the probability that the network requires more than M communication rounds to compute MR' iterations of (17). Further suppose that the packet erasure probability is p and that erasures are

symmetric. Then

$$P_{M,R'} \leq N2^{-M(D(1-R',p)-\log(\Delta+1))} \quad (19)$$

In particular, whenever R' satisfies

$$D(1 - R', p) > \log(\Delta + 1) \quad (20)$$

$P_{M,R'}$ decays exponentially fast in M . Recall that N is the number of nodes and Δ the maximum degree.

Proof: See Appendix C. ■

Using Theorem 7.1, we can determine the convergence rate Algorithm 1, μ_c^s , and it is given by

$$\mu_c^s \leq \mu^{R'} \quad (21)$$

where R' is the largest rate such that (20) is satisfied and μ is defined in (7). The superscript and subscript in μ_c^s denote that it is the convergence rate with coding under symmetric erasures. We will compare this with the convergence rate without coding in Section VIII. Let

$$R(p) \triangleq \sup_{R' \geq 0} \{R' \mid D(1 - R', p) > \log(\Delta + 1)\} \quad (22)$$

Then it is easy to see that $R(p) > 0$ if and only if $p < 1/(1 + \Delta)$. This means that the proof technique used here does not allow us to prove average consensus if the erasure probability is larger than $1/(1 + \Delta)$. We can demonstrate how to overcome this. In fact, one can show that average consensus will be achieved for all $0 \leq p \leq 1$, we will state the result as follows.

Theorem 7.2: Let $P_{M,R'}$ denote the probability that the network requires more than M communication rounds to compute MR' iterations of (17). Further suppose that the packet erasure probability is p and that erasures are symmetric. Then

$$P_{M,R'} \leq N2^{-MD(R', (1-p)^{|\mathcal{E}|})} \quad (23)$$

In particular, whenever R' satisfies

$$R' < (1 - p)^{|\mathcal{E}|} \quad (24)$$

$P_{M,R'}$ decays exponentially fast in M . Recall that N is the number of nodes and $|\mathcal{E}|$ is the number of edges in the network.

Proof: See Appendix E ■

Combining Theorems 7.1 and 7.2, we conclude that the convergence rate of Algorithm 1, μ_c^s , is given by

$$\mu_c^s \leq \min\{\mu^{R(p)}, \mu^{(1-p)^{|\mathcal{E}|}}\} \quad (25)$$

B. Asymmetric Link Failures and Tree Codes

Now suppose packet erasures are not symmetric. Since information at each node is generated one packet at a time and since the unit of communication is a packet, the rate of the code is $R = 1/n^1$. Here, one round of communication corresponds to every pair of neighbors exchanging n packets each. Then in any communication round, node i does not know which of the n transmitted packets have been received by each of its neighbors. In this case, we use the anytime reliable codes described in Section VI-A.

1) *The protocol:* Consider the pair of nodes i, j and let b_t^{ji} denote the t^{th} information packet destined to node j from node i . Then the data actually transmitted by node i is given by

$$c_\ell^{ji} = \sum_{\ell'=1}^{\ell} G_{\ell'} b_{\ell'}^{ji} \quad (26)$$

Since the code is (R, β) -anytime reliable, we have $P(\hat{b}_{\ell|\ell}^{ji} \neq b_{\ell}^{ji}) \leq 2^{-\beta(\ell-\ell')}$. Since the channel is an erasure channel, the maximum likelihood decoder amounts to solving linear equations. This can be done recursively and efficiently as shown in (our paper). Whenever the equations admit a unique solution to some of the variables, those variables are correctly decoded. We leave the remaining variables as erasures and do not venture a guess about their value. As a result, the decoder always knows whenever it decodes something correctly.

Like in the case of repetition coding for symmetric erasures, for each link $j \rightarrow i$, we associate two queues $Q_{in,t}^{ij}$ and $Q_{out,t}^{ij}$ although with a slightly different meaning. The queue $Q_{in,t}^{ij}$ contains all the information packets transmitted by node j to node i till round t . In other words, $Q_{in,t}^{ij} = \{b_\tau^{ij}\}_{\tau \leq t}$. On the other hand, $Q_{out,t}^{ij}$ are node i 's estimates of the information packets transmitted by node j so far, i.e., $Q_{out,t}^{ij} = \{\hat{b}_{\tau|t}^{ij}\}_{\tau \leq t}$. Also, it will be evident from Algorithm 2 that $Q_{in,t}^{ji} = Q_{in,t}^{j'i}$ for all $j, j' \in \mathcal{N}_i$.

With this setup, the mechanics of the protocol is very simple and is outlined in Algorithm 2

We can now compute the convergence rate of average consensus achieved by the above algorithm and we state it as the following Theorem.

Theorem 7.3: Let $P_{M,R'}$ denote the probability that the network requires more than M communication rounds to compute MR' iterations of (17). Further suppose that the packet erasure probability is p and that erasures are

¹This kind of rate is because we are quantizing each number x_k^i to fit into one packet. One can instead quantize it more finely into multiple packets, say k , in which case $R = k/n$

Algorithm 2 Node i 's transmission to its neighbors in round t

- 1: For each $j' \in \mathcal{N}_i$, compute $\ell_{t,j'} = \max\{\ell' \mid x_{\ell'}^{j'} \in Q_{out,t}^{ij'}\}$ and let $\ell_t = \min_{j' \in \mathcal{N}_i} \ell_{t,j'}$
 - 2: Also compute $m_{t,j'} = \max\{m' \mid x_{m'}^i \in Q_{in,t}^{ji'}\}$ and let $m_t = \min_{j' \in \mathcal{N}_i} m_{t,j'}$
 - 3: **if** $\ell_t + 1 > m_{t-1}$ **then**
 - 4: Compute $x_{m_{t-1}+1}^i$ using (17) and set $b_t^{ji} = x_{m_{t-1}+1}^i$ for all $j \in \mathcal{N}_i$
 - 5: **else**
 - 6: set $b_t^{ji} = w$ for all $j \in \mathcal{N}_i$
 - 7: **end if**
-

asymmetric. Suppose each node uses a (R, β) –anytime reliable code. Then

$$P_{M,R'} \leq N 2^{-M((1-R')\frac{\beta}{2} - H(R') - \log(\Delta+1))} \quad (27)$$

In particular, whenever R' satisfies

$$(1 - R')\beta/2 > H(R') + \log(\Delta + 1) \quad (28)$$

$P_{M,R'}$ decays exponentially fast in M .

Proof: See Appendix D ■

As in the symmetric case, the convergence rate, μ_c^a , using tree codes is given by

$$\mu_c^a = \mu^{R'} \quad (29)$$

where R' is the largest rate such that (28) is satisfied and μ is as in (7). Let

$$R(\beta) \triangleq \sup_{R' \geq 0} \{R' \mid (1 - R')\beta/2 > H(R') + \log(\Delta + 1)\} \quad (30)$$

Then much like in Section VII-A, it is easy to see that $R(\beta) > 0$ if and only if $\beta > 2 \log(1 + \Delta)$.

VIII. DISCUSSION - CODING VS NO CODING

When there is no coding, the consensus recursion is given by (9). We begin with the case of symmetric erasures.

A. Symmetric Erasures

The convergence rate of (9) when erasures are symmetric is given by the following Lemma

Lemma 8.1 (Symmetric Erasures): When the erasures are symmetric and i.i.d over time and space, the convergence rate of (9), μ_c^s which we define as

$$\mu_c^s = \sup_{x_o \neq r\mathbf{1}} \lim_{k \rightarrow \infty} \left[\frac{\mathbb{E} \|x_k - r\mathbf{1}\|^2}{\|x_o - r\mathbf{1}\|^2} \right]^{\frac{1}{2k}} \quad (31)$$

is given by

$$\mu_{\bar{c}}^s = \sqrt{\lambda_2(\Gamma_s)} \quad (32)$$

where $\Gamma_s = \mathbb{E}(I - \epsilon\mathcal{L}_0) \otimes (I - \epsilon\mathcal{L}_0)$ is a deterministic matrix that is a function of ϵ, p, \mathcal{L} and can be computed explicitly in closed form. The subscript \bar{c} indicates that there is no coding and the subscript s in Γ_s is because the erasures are symmetric

Proof: See Appendix A. ■

Consider the case of coding in the presence of symmetric erasures. From Theorem 7.1 and (8), it is easy to see that the convergence rate is given by μ_c^s in (21). So, whenever $\mu_c^s < \mu_{\bar{c}}^s$, coding offers an advantage. We state this as a Theorem

Theorem 8.2: In the case of symmetric erasures, coding offers a faster convergence than (9) whenever there is a $R' > 0$ such that

$$(1 - R') \log \frac{1}{p} > \log(\Delta + 1) + H(R') \quad (33a)$$

$$\left(\rho \left(I - \epsilon\mathcal{L} - \frac{1}{N} \mathbf{1}\mathbf{1}^T \right) \right)^{R'} < \sqrt{\lambda_2(\Gamma_s)} \quad (33b)$$

B. Asymmetric Erasures

As mentioned in Section V, when link failures are asymmetric, the algorithm of (9) does not achieve average consensus. Nevertheless the nodes reach agreement and the rate of convergence to agreement has been characterized in [14]. Here, we characterize the mean squared error of the state from average consensus.

Lemma 8.3 (Asymmetric Erasures): When the erasures are asymmetric and i.i.d over time and space, we have

$$\mathbb{E}\|x_k - r\mathbf{1}\|^2 = (x_o - r\mathbf{1})^T \otimes (x_o - r\mathbf{1})^T \Gamma_a^k \text{vec}(I) \quad (34)$$

Here I is an $N \times N$ identity matrix and

$$\Gamma_a = \mathbb{E}(I - \epsilon\mathcal{L}_0^T) \otimes (I - \epsilon\mathcal{L}_0^T) \quad (35)$$

where Γ_a is a deterministic matrix that is a function of ϵ, p, \mathcal{L} and can be computed explicitly in closed form. Furthermore $\rho(\Gamma_a) = 1$.

Proof: See Appendix B. ■

Note that $\mathbf{1}^T \Gamma_a = \mathbf{1}^T$ but $\Gamma_a \mathbf{1} \neq \mathbf{1}$. Let $c, \|c\| = 1$ be the right eigen vector of Γ_a corresponding to eigen value 1, i.e., $\Gamma_a c = c$. Then, it is easy to see that $\lim_{k \rightarrow \infty} \Gamma_a^k = \frac{1}{N} c \mathbf{1}^T$. Using this in (34), we get

$$\lim_{k \rightarrow \infty} \mathbb{E} \|x_k - r \mathbf{1}\|^2 = (x_o - r \mathbf{1})^T \otimes (x_o - r \mathbf{1})^T c \quad (36)$$

This proves that one cannot achieve average consensus without coding when link failures are asymmetric. So, a major benefit of using tree codes in such cases is to *guarantee* average consensus. Furthermore, tree codes can be used to implement *any* distributed protocol over a network with erasure links.

APPENDIX

A. Proof of Lemma 8.1

Note that $\mathcal{L}_k \mathbf{1} = 0$ whether or not the erasures are symmetric. Recall that $r = \frac{1}{N} \mathbf{1}^T x_0$.

$$x_k - r \mathbf{1} = (I - \epsilon \mathcal{L}_{k-1})(x_{k-1} - r \mathbf{1}) \quad (37a)$$

$$x_k - r \mathbf{1} = \quad (37b)$$

$$\underbrace{(I - \epsilon \mathcal{L}_{k-1})(I - \epsilon \mathcal{L}_{k-2}) \dots (I - \epsilon \mathcal{L}_0)}_{\triangleq Y_k} (x_0 - r \mathbf{1}) \quad (37c)$$

$$\begin{aligned} \mathbb{E} \|x_k - r \mathbf{1}\|^2 &= (x_0 - r \mathbf{1})^T \mathbb{E} Y_k^T Y_k (x_0 - r \mathbf{1}) \\ &= (x_0 - r \mathbf{1})^T \otimes (x_0 - r \mathbf{1})^T \text{vec}(P_k) \end{aligned} \quad (38)$$

where $P_k = \mathbb{E} Y_k^T Y_k$. Recall that the erasure process is independent over time and across links. Then we have

$$P_k = \mathbb{E}(I - \epsilon \mathcal{L}_0^T) P_{k-1} (I - \epsilon \mathcal{L}_0) \quad (39a)$$

$$\text{vec}(P_k) = \Gamma_s \text{vec}(P_{k-1}), \text{ where} \quad (39b)$$

$$\Gamma_s = \mathbb{E}(I - \epsilon \mathcal{L}_0^T) \otimes (I - \epsilon \mathcal{L}_0^T) \quad (39c)$$

Since erasures are symmetric, $\mathcal{L}_0^T = \mathcal{L}_0$. Furthermore, we have $\text{vec}(P_k) = \Gamma_s^k \text{vec}(I)$, where I is an $N \times N$ identity matrix. Putting (38) and (39) together, we get

$$\mathbb{E} \|x_k - r \mathbf{1}\|^2 = (x_0 - r \mathbf{1})^T \otimes (x_0 - r \mathbf{1})^T \Gamma_s^k \text{vec}(I) \quad (40)$$

So, the rate of convergence of the consensus algorithm in the absence of coding is clearly determined by Γ_s . Observe that Γ_s is doubly stochastic, i.e., $\mathbf{1}^T \Gamma_s = \mathbf{1}^T$ and $\Gamma_s \mathbf{1} = \mathbf{1}$. It has one eigen value at 1 and all others are strickly

smaller than 1 in magnitude. Let $\lambda_2(\Gamma_s)$ denote the second largest eigen value in magnitude. Then clearly

$$\lim_{k \rightarrow \infty} \Gamma_s^k = \frac{1}{N^2} \mathbf{1}\mathbf{1}^T \quad (41)$$

and the rate of convergence is given by

$$\mu_{\bar{c}}^s = \sqrt{\lambda_2(\Gamma_s)} \quad (42)$$

B. Proof of Lemma 8.3

Except the claim $\rho(\Gamma_a) = 1$, everything else follows from Appendix A. Since $\Gamma_a = \mathbb{E}(I - \epsilon \mathcal{L}_0^T) \otimes (I - \epsilon \mathcal{L}_0^T)$, the claim $\rho(\Gamma_a) = 1$ follows if $\rho(I - \epsilon \mathcal{L}_0) = 1$ which is what we show. Recall that the random variable X_0^{ij} is defined as $X_0^{ij} = 0$ if the link $j \rightarrow i$ is erased at time 0 and $X_0^{ij} = 1$ otherwise. For brevity, we will write X^{ij} instead of X_0^{ij} . Then it is easy to verify that one can write \mathcal{L}_0 as follows

$$\mathcal{L}_0 = \sum a_{ij} X^{ij} e_i (e_i - e_j)^T \quad (43)$$

where e_i is the i^{th} unit vector. In particular, the underlying Laplacian in the absence of any erasures can be written as $\mathcal{L} = \sum a_{ij} e_i (e_i - e_j)^T$. For any $x \in \mathbb{R}^N$, we have

$$\begin{aligned} x^T (I - \epsilon \mathcal{L}_0) x &= x^T \left(I - \frac{\epsilon}{2} (\mathcal{L}_0 + \mathcal{L}_0^T) \right) x \\ &= \|x\|^2 - \frac{\epsilon}{2} \sum a_{ij} X^{ij} (x_i - x_j)^2 \leq \|x\|^2 \end{aligned} \quad (44)$$

Furthermore,

$$\begin{aligned} \|x\|^2 - \frac{\epsilon}{2} \sum a_{ij} X^{ij} (x_i - x_j)^2 &\geq \|x\|^2 - \frac{\epsilon}{2} \sum a_{ij} (x_i - x_j)^2 \\ &= x^T (I - \epsilon \mathcal{L}) x \geq -\|x\|^2 \end{aligned} \quad (45)$$

The last inequality follows from the fact that $\rho(I - \epsilon \mathcal{L}) = 1$. Combining (44) and (45), we have $|x^T (I - \epsilon \mathcal{L}_0) x| \leq \|x\|^2$ for all $x \in \mathbb{R}^N$ which implies that $\rho(I - \epsilon \mathcal{L}_0) \leq 1$. But $\mathcal{L}_0 \mathbf{1} = \mathbf{1}$, so $\rho(I - \epsilon \mathcal{L}_0) = 1$. Therefore $\rho(\Gamma_a) = 1$. This completes the proof.

C. Proof of Theorem 7.1

We will begin by identifying the state of the protocol in Algorithm 1. For the sake of clarity, we will refer to nodes using letters u, v , etc., instead of i, j . Recall that \mathcal{N}_v denotes the set of neighbors of v . For each node v at time t (i.e., after round t), we associate $|\mathcal{N}_v|$ variables $\{n_{vu}(t)\}_{u \in \mathcal{N}_v}$, where $n_{vu}(t)$ denotes the latest iterate of node u that

is available to node v at time t . In other words, $n_{vu}(t)$ is the largest integer τ such that x_τ^u is available to node v . We further define

$$n_v(t) \triangleq 1 + \min_{u \in \mathcal{N}_v} n_{vu}(t) \quad (46)$$

Note that $n_v(t)$ is the latest iteration of (17) that node v can compute at time t . In other words, node v has computed $\{x_\tau^v\}_{\tau \leq n_v(t)}$ and no more. With this setup, it is clear that Algorithm 1 would have executed $\min_v n_v(t)$ iterations of (17) till time t . Note that the rate of the protocol is then given by $R = \lim_{t \rightarrow \infty} \frac{\min_v n_v(t)}{t}$, which is a random variable for a specific run of the protocol. We now state the evolution of $n_{vu}(t)$ as a Lemma below.

Lemma 1.1: Let $X_t^{vu} = 1$ if the edge (v, u) is erased in round t and 0 otherwise. Then the evolution of $n_{vu}(t)$ is given by the following equation

$$n_{vu}(t+1) = n_{vu}(t) + X_{t+1}^{vu} \mathbb{1}_{[n_u(t) > n_{vu}(t)]} \quad (47)$$

Proof: The proof follows from the following simple observations

- 1) $n_{vu}(t)$ increases by atmost 1 in each step
- 2) In any round, if node u receives an erasure on a link, it will infer that its transmission on that link was also erased. As a result, node u has knowledge of $n_{vu}(t)$ at all times t
- 3) In round $t+1$, if either the edge (v, u) is erased or node u sends a w to node v , then $n_{vu}(t+1) = n_{vu}(t)$
- 4) Node u sends a ‘wait’ w to node v in round $t+1$ if and only if $n_{vu}(t) = n_u(t)$.

■

We say that round t got wasted at node v if $n_v(t-1) = n_v(t)$, i.e., node v could not perform a new iteration of (17) at time t . The proof idea is as follows: for each node v at time t , we will argue that there exists a sequence of t edges of which at least $t - n_v(t)$ edges have failed. We then union bound over all possible choices of such t edges.

Before proceeding further, we define an object which we call the ‘trellis’, for lack of a better word. Associated to any undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ represented by the adjacency matrix A , we define an infinite trellis $\mathcal{T}(\mathcal{G}) = (\mathcal{V}_{\mathcal{T}}, \mathcal{E}_{\mathcal{T}})$ as follows. Associated to each node v in \mathcal{V} , there are countably infinitely many copies $\{v_k\}_{k \geq 0}$ in $\mathcal{V}_{\mathcal{T}}$. Let I denote a $|\mathcal{V}| \times |\mathcal{V}|$ identity matrix. Then the nodes $\mathcal{V}_{\mathcal{T}}$ and edges $\mathcal{E}_{\mathcal{T}}$ of $\mathcal{T}(\mathcal{G})$ are given by

$$\mathcal{V}_{\mathcal{T}} = \bigcup_{v \in \mathcal{V}} \bigcup_{k \geq 0} \{v_k\} \quad (48a)$$

$$\mathcal{E}_{\mathcal{T}} = \{(v_\tau, u_{\tau'}) \mid |\tau - \tau'| = 1, (A + I)_{vu} = 1\} \quad (48b)$$

The edges in $\mathcal{E}_{\mathcal{T}}$ are all undirected, i.e., (u_0, v_1) and (v_1, u_0) are treated as a single edge. The trellis for an example network is given in Fig 2.

Definition 2 (time-like): Any sequence of edges (or a path), \mathcal{S}_t , in the trellis $\mathcal{T}(\mathcal{G})$ of the type

$$\mathcal{S}_t = \left\{ (v_t, u_{t-1}^{(t-1)}), (u_{t-1}^{(t-1)}, u_{t-2}^{(t-2)}), \dots, (u_1^{(1)}, u_0^{(0)}) \right\}$$

will be called ‘time-like’ ending in node v_t

An edge $(u_\tau^{(\tau)}, u_{\tau-1}^{(\tau-1)}) \in \mathcal{E}_{\mathcal{T}}$ is said to be erased if there was an erasure on the edge $(u^{(\tau)}, u^{(\tau-1)}) \in \mathcal{E}$ in round τ .

The time-like sequence \mathcal{S}_t is said to have ℓ erasures if ℓ of the t edges in \mathcal{S}_t were erased. We are now ready to state the key Lemma from which the proof of Theorem 7.1 follows easily.

Lemma 1.2: If after t rounds of communication, node v has performed $n_v(t)$ iterations of (17), then there exists a time-like sequence of t edges ending in node v_t that have at least $t - n_v(t)$ erasures among them.

We will first prove Theorem 7.1 using Lemma 1.2. Suppose after t communication rounds, node v performed Rt iterations of (17), for some $R < 1 - p$. Recall that the probability of an erasure is p . Then there must be a time like sequence of t edges with at least $(1 - R)t$ erasures, the probability of which is approximately $2^{-tD(1-R,p)}$, where $D(q, p) = q \log(q/p) + (1 - q) \log(1 - q/1 - p)$. Now there are at most $(\Delta + 1)^t$ choices of such time-like sequences. Then, doing a union bound over all these sequences, we get

$$P_{R,t} \leq N(\Delta + 1)^t 2^{-tD(1-R,p)} \quad (49)$$

where $P_{R,t}$ is the probability that the network performed Rt or fewer iterations of (17) in t rounds and N is the number of nodes in the network. This is the claim in Theorem 7.1. We will now prove the Lemma.

Proof: [Proof of Lemma 1.2]

For ease of presentation, we will introduce the following notation in the rest of the proof.

- a) we will refer to any time-like sequence of τ edges ending in v_τ that has $\tau - n_v(\tau)$ or more erasures as a “witness” at v_τ .
- b) We will call a node $u \in \mathcal{N}_v$ a “bottleneck” for node v in round t iff $n_{vu}(t - 1) = n_v(t - 1) - 1$, i.e., $n_{vu}(t - 1) = \min_{u' \in \mathcal{N}_v} n_{vu'}(t - 1)$.

The Lemma claims that there is a witness at v_t for all $v \in \mathcal{V}$ and $t \geq 0$. We will prove this by induction. The hypothesis is clearly true for $t = 0$. Suppose it is true for all nodes $v \in \mathcal{V}$ and all $\tau \leq t - 1$. Recall that we say that round t at node v is wasted only if $n_v(t - 1) = n_v(t)$. There are two broad cases, round t gets wasted at node v or it does not.

- 1) Suppose round t is not wasted, i.e., $n_v(t) = n_v(t - 1) + 1$. Then by the induction hypothesis, there is a witness at v_{t-1} . Appending the edge (v_{t-1}, v_t) to this witness gives us a witness for v_t .
- 2) It remains to consider the case where round t gets wasted at node v , i.e., $n_v(t) = n_v(t - 1)$.

We will divide case 2) above into two sub-cases: a) \exists a $u \in \mathcal{N}_v$ s.t $n_u(t-1) = n_v(t-1) - 1$ and b) such a neighbor does not exist.

- a) If there is a neighbor $u \in \mathcal{N}_v$ such that $n_u(t-1) = n_v(t-1) - 1$, then the witness for v_t is obtained by appending the edge (v_t, u_{t-1}) to the witness at u_{t-1} .
- b) Here $n_u(t-1) \geq n_v(t-1)$ for all $u \in \mathcal{N}_v$. Since $|n_u(\tau) - n_v(\tau)| \leq 1$ for any τ , we can partition the neighbors of v into two classes $Y = \{u \in \mathcal{N}_v \mid n_u(t-1) = n_v(t-1)\}$ and $Z = \{u \in \mathcal{N}_v \mid n_u(t-1) = n_v(t-1) + 1\}$. Furthermore, let $B = \{u \in \mathcal{N}_v \mid n_{vu}(t-1) = n_v(t-1) - 1\}$ denote the bottlenecks for v in round t .

We will further divide case b) above into two sub-cases: i) $B \cap Z = \emptyset$ and ii) $B \cap Z \neq \emptyset$

- i) $B \cap Z = \emptyset$, i.e., there are no bottlenecks in the set of neighbors Z . Observe that a bottleneck neighbor will not send a wait w . Also for any $u \in B \cap Y$, $n_{vu}(t-1) = n_v(t-1) - 1 = n_u(t-1) - 1$. So, the data transmitted by node u to node v in round t is $x_{n_u(t)}^u$, i.e., iteration $n_u(t)$ of (17). Since round t at node v got wasted, at least one of the edges to a bottleneck neighbor must have been erased in round t . Otherwise, node v would have been able to compute a new iteration of (17) and the round would not have been wasted. Suppose the erasure happened on edge (v, u) for some $u \in B \cap Y$. Then appending edge (v_t, u_{t-1}) to the witness at u_{t-1} will give us the witness at v_t .
- ii) $B \cap Z \neq \emptyset$, i.e., there is a neighbor $u \in B \cap Z$ such that $n_u(t-1) = n_v(t-1) + 1$ and $n_{vu}(t-1) = n_v(t-1) - 1 = n_u(t-1) - 2$. Furthermore, there must be a neighbor $u \in B \cap Z$ whose transmission to v in round t must have been erased (else there must be an edge to $B \cap Y$ which was erased and we revert back to case i)). Note that $n_u(t-2) \geq n_v(t-1)$. It follows from Lemma 1.1 that node u must have transmitted iteration $n_v(t-1)$ in round $t-1$ as well as round t and both were erased since $n_{vu}(t) = n_{vu}(t-1) = n_v(t-1) - 1$. Since this erasure model considers symmetric erasures, the transmission from v to u in round $t-2$ is also erased. Appending the edges (v_t, u_{t-1}) and (u_{t-1}, v_{t-2}) to the witness at v_{t-2} gives us the witness for v_t .

This completes the proof of Lemma 1.2. ■

D. Proof of Theorem 7.3

We will begin the proof with three preliminary results before moving to the main argument. Recall that an (R, β) -anytime reliable code is one that guarantees $P(\hat{b}_{\tau|t} \neq b_\tau) \leq 2^{-\beta(t-\tau+1)}$. For such a code that is linear, we can say the following.

Lemma 1.3: Suppose $\{b_i\}_{i \geq 0}$ are encoded and decoded using a causal linear (R, β) -anytime reliable code. Consider the following events, $Y(\tau'_1, \tau_1): \tau'_1 = 1 + \arg\max_\ell \{\hat{b}_{\ell|\tau_1} = b_\ell\}$ and $Y(\tau'_2, \tau_2): \tau'_2 = 1 + \arg\max_\ell \{\hat{b}_{\ell|\tau_2} = b_\ell\}$, i.e.,

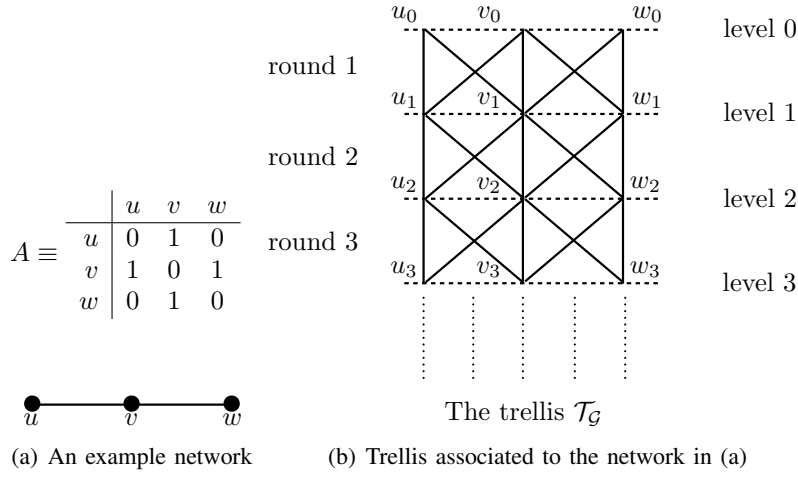


Fig. 2. This depicts the trellis associated to a network of three nodes connected in a straight line. The thick lines represent edges.

$Y(\tau'_i, \tau_i)$ is the event that at decoding instant τ_i , the position of the earliest error is at τ'_i for $i = 1, 2$. Furthermore, suppose that the intervals $[\tau'_1, \tau_1]$ and $[\tau'_2, \tau_2]$ are disjoint. Then we have

$$P(Y(\tau'_1, \tau_1) \cap Y(\tau'_2, \tau_2)) \leq 2^{-\beta(|\tau_1 - \tau'_1| + 1 + |\tau_2 - \tau'_2| + 1)} \quad (50)$$

The probability above is only over the randomness of the channel.

Proof: Without loss of generality, assume that $\tau'_2 > \tau_1$. Due to linearity, we can assume without losing generality that the input $b_i = 0$ for $i \geq 0$. Let E_i denote the portion of the erasure pattern introduced by the channel during the interval $[\tau'_i, \tau_i]$ that resulted in the event $Y(\tau'_i, \tau_i)$. Then, we claim that $P(E_i) \leq 2^{-\beta|\tau_i - \tau'_i| + 1}$. This follows from the simple observation that if the encoder input in the first $\tau_i - \tau'_i + 1$ instants is all zero and the corresponding channel erasure pattern is E_i , then $Y(\tau'_i, \tau_i)$ implies that at the decoding instant $\tau_i - \tau'_i$, the earliest error would have happened at time 0, the probability of which is at most $2^{-\beta|\tau_i - \tau'_i| + 1}$.

Since the intervals $[\tau'_1, \tau_1]$ and $[\tau'_2, \tau_2]$ are disjoint, the erasure patterns E_1 and E_2 correspond to independent channel uses. So we have

$$P(Y(\tau'_1, \tau_1) \cap Y(\tau'_2, \tau_2)) \leq P(E_1, E_2) = P(E_1)P(E_2)$$

The result now follows. ■

For ease of presentation, we introduce the following definition

Definition 3 (Error Interval): With respect to the notation in Lemma 1.3, we refer to the interval $[\tau'_i, \tau_i]$ as the error at time τ_i .

Before proceeding with the rest of the proof, we will recall a Lemma from [15] and state it here for easy reference.

Lemma 1.4 (Lemma 7, [15]): In any finite set of intervals on the real line whose union J is of total length s there is a subset of disjoint intervals whose union is of total length at least $s/2$

We will now state a version of Lemma 1.3 when the error intervals are not necessarily disjoint.

Lemma 1.5: If $\{b_i\}_{i \geq 0}$ are encoded and decoded using a causal linear (R, β) –anytime reliable code, then

$$P\left(\hat{b}_{\tau'_1|\tau_1} \neq b_{\tau'_1}, \dots, \hat{b}_{\tau'_m|\tau_m} \neq b_{\tau'_m}\right) \leq 2^{-\frac{\beta(\sum_i |\tau_i - \tau'_i| + 1)}{2}}$$

Proof: The proof follows directly from Lemma 1.3 and Lemma 1.4. ■

We use an argument very similar to the one used in proving Theorem 7.1. We will define a trellis $\vec{\mathcal{T}}(\mathcal{G})$ exactly the same way we defined $\mathcal{T}(\mathcal{G})$ except that the edges $\vec{\mathcal{E}}_{\mathcal{T}}$ are now directed and they point forward in time, i.e., downwards w.r.t to the Fig 2(b). In other words, for neighbors $(u, v) \in \mathcal{V}$, the edge (v_t, u_{t-1}) is directed from node u_{t-1} to node v_t and represents the transmission from u to v in round t .

Recall the definition of a time-like sequence of edges, \mathcal{S}_t , from Definition 2. Let

$$\mathcal{S}_t = \left\{ (u_t^{(t)}, u_{t-1}^{(t-1)}), (u_{t-1}^{(t-1)}, u_{t-2}^{(t-2)}), \dots, (u_1^{(1)}, u_0^{(0)}) \right\}$$

Let B_τ be the error interval at decoding instant τ on the edge node $(u^{(\tau)}, u^{(\tau-1)}) \in \mathcal{E}$. We alternately call B_τ the error interval on the edge $(u_\tau^{(\tau)}, u_{\tau-1}^{(\tau-1)}) \in \vec{\mathcal{E}}_{\mathcal{T}}$. Then we define $|\mathcal{S}_t|$ as follows

$$|\mathcal{S}_t| = \sum_{(v,u) \in \mathcal{E}} |B_{vu}|, \text{ where} \tag{51}$$

$$B_{vu} = \bigcup_{\tau: (u^{(\tau)}, u^{(\tau-1)}) = (v,u)} B_\tau \tag{52}$$

This definition is motivated by the fact that the packet erasure events during an error interval on a given edge, say $(v, u) \in \mathcal{E}$, are independent of those in an error interval on a different edge $(v', u') \neq (v, u)$ in any round of communication. So, intuitively $|\mathcal{S}_t|$ captures the number of independent “bad” channel realizations seen by the edges in \mathcal{S}_t . In what follows, we will show a connection between the number of wasted communication rounds at the node u^t and the number $|\mathcal{S}_t|$.

A *witness* at node v_t is a time like sequence of edges \mathcal{S}_t such that $|\mathcal{S}| \geq t - n_v(t)$. In Lemma 1.6, we will demonstrate a witness for v_t for all $v \in \mathcal{V}$ and $t \geq 0$. The technique is very similar to the proof of Lemma 1.2 and hence we will only provide a sketch of the proof. After that we will use Lemma 1.5 to prove that $P(t - n_v(t) \geq m) \leq (\Delta + 1)^t \binom{t}{m} 2^{-m\beta/2}$ for any $v \in \mathcal{V}$.

Lemma 1.6: If after t rounds of communication, node v has performed $n_v(t)$ iterations of (17), then there exists a time-like sequence, \mathcal{S}_t of t edges in $\vec{\mathcal{E}}_{\mathcal{T}}$ ending in node v_t with $|\mathcal{S}_t| > t - n_v(t)$

Proof: The proof is obtained by repeating the same argument as in the proof of Lemma 1.2 with the word ‘erasure’ replaced with the word ‘tree code error’. The only case that needs a little bit of clarification is case 2-b-ii, i.e., round t is wasted at node v and $B \cap Z \neq \emptyset$, where B and Z retain the same meaning as before. In this case, like before, there is a neighbor $u \in \mathcal{N}_v$ such that $n_{vu}(t) = n_u(t-1) - 2$. From Algorithm 2, it is clear that node the information $x_{n_u(t-1)-1}^u$ was encoded and transmitted by node u to node v in round $t-1$ or before. Therefore, the error interval on the edge $(v_t, u_{t-1}) \in \vec{\mathcal{E}}_{\mathcal{T}}$ contains the interval $[t-1, t]$. Let the witness at node u_{t-1} be $\mathcal{S}_{t-1,u}$. Append the edge (v_t, u_{t-1}) to $\mathcal{S}_{t-1,u}$ to get a new time-like sequence which we call $\mathcal{S}_{t,v}$. We claim that $\mathcal{S}_{t,v}$ is a witness at v_t . This proof of this claim follows from the following observations

- 1) When applying Lemma 1.5, we only to care about error intervals on the same edge at different times
- 2) The edge (v, u) appears in the time-like sequence $\mathcal{S}_{t,v}$ for round t and hence, it can possibly appear again only in $\mathcal{S}_{t,v}$ in round $t-2$ or earlier. So, the length of the union of the error intervals on the edge $(v_t, u_{t-1}) \in \mathcal{S}_{t-1,u}$ increases by at least 2 with the addition of the edge (v_t, u_{t-1}) . Hence we have

$$|\mathcal{S}_{t,v}| \geq |\mathcal{S}_{t-1,u}| + 2 \geq t-1 - n_u(t-1) = t - n_v(t)$$

This completes the proof of Lemma 1.6.

Putting together Lemma 1.6 and Lemma 1.5, we have

$$P(t - n_v(t) \geq m) \leq (\Delta + 1)^t \binom{t}{m} 2^{-\beta m/2}$$

The result now follows trivially. ■

E. Proof of Theorem 7.2

The bound $(1-p)^{|\mathcal{E}|}$ is intuitively motivated by the following observation, in a given round of communication, $(1-p)^{|\mathcal{E}|}$ is the probability that none of the edges are erased. As a result one would expect the fraction of communication rounds in which nodes can perform an iteration of (17) to be approximately $(1-p)^{|\mathcal{E}|}$. The above observation alone would not render a proof because successful communication could also mean that a node received only ‘waits’ from its neighbors and hence could not compute an iteration of (17). The proof idea is simple but conveying it requires some setup. Let $W_{uv}^{(t)}$ denote the event where node v transmits a ‘wait’ to node u in round t . We introduce the following definition

Definition 4: Consider nodes v, u, u' such that $u \in \mathcal{N}_v$ and $u' \in \mathcal{N}_u$. Also suppose that node v transmits a ‘wait’ to node u in round τ and node u transmits a ‘wait’ to node u' in round $\tau+1$, i.e., events $W_{uv}^{(\tau)}$ and $W_{u'u}^{(\tau+1)}$ happen. Then $W_{uv}^{(\tau)}$ is said to have caused $W_{u'u}^{(\tau+1)}$ if both the following conditions hold

$$(a) \ n_u(\tau - 1) = 1 + n_{uv}(\tau - 1)$$

$$(b) \ n_{u'u}(\tau) = n_u(\tau)$$

To understand the definition, observe that condition (a) implies that node v is a bottleneck node for node u in round τ and condition (b) implies that node u' already knows $n_u(\tau)$ after round τ . Node u could not perform a new iteration in round τ since it received a ‘wait’ from a bottleneck node (in this case v) and hence sent a ‘wait’ to node u' . So, it is natural to blame $W_{uv}^{(\tau)}$ for $W_{u'u}^{(\tau+1)}$. Note that Definition 4 is further justified by the observation that a ‘wait’ in round τ will either have an effect in round $\tau + 1$ or will never. Also note that Definition 4 can be extended to more than two waits by having conditions (a) and (b) hold for every pair of successive ‘wait’ events.

With that, we are now ready to state the main Lemma. The Lemma essentially implies that ‘waits’ do not loop in the network. In other words, if in round τ a node v transmits a ‘wait’, then this ‘wait’ will not *cause* the same node v to transmit another ‘wait’ in a future round $\tau' > \tau$.

Lemma 1.7 (‘Waits’ do not loop): Consider the sequence of events $\{W_{u_{i+1}u_i}^{(\tau+i-1)}\}_{i=1}^{\ell}$ such that $W_{u_{i+1}u_i}^{(\tau+i-1)}$ is caused by $W_{u_i u_{i-1}}^{(\tau+i-2)}$ for all $2 \leq i \leq \ell$. Then the nodes $\{u_i\}_{i=1}^{\ell}$ are all distinct.

Proof: Node u_1 sent a ‘wait’ to node u_2 in round τ implies that $n_{u_1}(\tau - 1) = n_{u_2 u_1}(\tau - 1)$. Furthermore, since $W_{u_3 u_2}^{(\tau+1)}$ is caused by $W_{u_2 u_1}^{(\tau)}$, conditions (a) and (b) in Definition 4 apply. In particular, condition (a) together with the first observation gives $n_{u_2}(\tau - 1) = n_{u_1}(\tau - 1) + 1$. Since node u_2 could not perform a new iteration of (17), we have $n_{u_2}(\tau) = n_{u_2}(\tau - 1) = n_{u_1}(\tau - 1) + 1$. Repeating this argument for the remaining nodes, we get

$$n_{u_{i+1}}(\tau + i - 1) = n_{u_i}(\tau_i - 2), \forall 1 \leq i \leq \ell \quad (53)$$

Now suppose the nodes $\{u_i\}_{i=1}^{\ell}$ are not all distinct. In particular, suppose $u_{\ell} = u_1$. Then from (53), we have $n_{u_1}(\tau + \ell - 2) = n_{u_{\ell}}(\tau + \ell - 2) = \ell - 1 + n_{u_1}(\tau - 1)$ which is not possible since $n_{u_1}(\cdot)$ can increment by at most 1 in each round and $n_{u_1}(\tau) = n_{u_1}(\tau - 1)$.

One will similarly arrive at a contradiction if any other node repeats in $\{u_i\}_{i=1}^{\ell}$. ■

The implication of Lemma 1.7 is clear. If a node v sends a ‘wait’ in round τ to any of its neighbors, then this ‘wait’ will not by itself stop node v from performing an iteration of (17) in a future round.

We are now ready to provide the main argument. Let $d(v, u)$ denote the length of the shortest path from node u to node v . So, if $v \in \mathcal{N}_u$, then $d(v, u) = 1$ and $d(v, v) = 0$. Let the diameter of the graph be δ , i.e., $\delta = \max_{u, v \in \mathcal{V}} d(v, u)$. And for an edge $e_{uu'} \equiv (u, u') \in \mathcal{E}$, we define

$$d(v, e_{uu'}) = \min\{d(v, u), d(v, u')\}$$

Let $\mathcal{E}_v^{(i)} \triangleq \{e \in \mathcal{E} \mid d(v, e) = i\}$. In view of Lemma 1.7, it is not difficult to see that an erasure on an edge in $\mathcal{E}_v^{(i)}$ in round τ will have an effect (if any) at node v only in round $\tau + i$. Let $A_{i,\tau}$ denote the event that there is an erasure on an edge in $\mathcal{E}_v^{(i)}$ in round τ . Then for $\tau \geq \delta$, it is easy to see that $\cap_{i=0}^{\delta} A_{i,\tau-i}^c$ implies that the round τ at node v is *not* wasted, i.e., node v can compute an iteration of (17). In other words

$$P(n_v(\tau) = n_v(\tau - 1)) \leq 1 - P\left(\bigcap_{i=0}^{\delta} A_{i,\tau-i}^c\right) = 1 - (1 - p)^{|\mathcal{E}|}$$

Due to the erasure model, note that the event $A_{i,\tau}$ is independent of $A_{i',\tau'}$ for $(i, \tau) \neq (i', \tau')$. Let

$$X_\tau = \mathbb{1}_{[n_v(\tau) = n_v(\tau-1)]}$$

$$Y_\tau = \mathbb{1}_{[\cup_{i=0}^{\delta} A_{i,\tau-i}]}$$

Then from the above argument $X_\tau = 1$ implies $Y_\tau = 1$ and $\{Y_\tau\}$ are independent Bernoulli random variables. Note that $P(Y_\tau = 1) \leq 1 - (1 - p)^{|\mathcal{E}|}$. Let $R' = \frac{n_v(t)}{t}$, then we have

$$P(t - n_v(t) = m) = P\left(\sum_{\tau=0}^t X_\tau = m\right) \leq P\left(\sum_{\tau=0}^t Y_\tau \geq m\right) \leq 2^{-tD(1-R', 1-(1-p)^{|\mathcal{E}|})} = 2^{-tD(R', (1-p)^{|\mathcal{E}|})}$$

The last inequality follows from a standard Chernoff bounding technique and is true whenever $R' < (1 - p)^{|\mathcal{E}|}$. Union bounding over all nodes $v \in \mathcal{V}$, we have

$$P(\exists v \in \mathcal{V} \ni n_v(t) \leq R't) \leq N 2^{-tD(R', (1-p)^{|\mathcal{E}|})}$$

This completes the proof.

REFERENCES

- [1] R. Olfati-Saber and R.M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *Automatic Control, IEEE Transactions on*, vol. 49, no. 9, sept. 2004.
- [2] R. Olfati-Saber, J.A. Fax, and R.M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, jan. 2007.
- [3] A. Jadbabaie, Jie Lin, and A.S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *Automatic Control, IEEE Transactions on*, vol. 48, no. 6, june 2003.
- [4] John N Tsitsiklis, "Problems in decentralized decision making and computation," 1984.
- [5] J. Tsitsiklis, D. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *Automatic Control, IEEE Transactions on*, vol. 31, no. 9, sep 1986.
- [6] Wei Ren and R.W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *Automatic Control, IEEE Transactions on*, vol. 50, no. 5, may 2005.
- [7] R.T. Sukhvasi and B. Hassibi, "Linear error correcting codes with anytime reliability," in *2011 IEEE International Symposium on Information Theory Proceedings (ISIT)*, aug 2011.

- [8] Lin Xiao and Stephen Boyd, "Fast linear iterations for distributed averaging," *Systems and Control Letters*, vol. 53, no. 1, 2004.
- [9] L. Moreau, "Stability of multiagent systems with time-dependent communication links," *Automatic Control, IEEE Transactions on*, vol. 50, no. 2, feb. 2005.
- [10] Y. Hatano and M. Mesbahi, "Agreement over random networks," *Automatic Control, IEEE Transactions on*, vol. 50, no. 11, nov. 2005.
- [11] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *Information Theory, IEEE Transactions on*, vol. 52, no. 6, june 2006.
- [12] Sridhar Rajagopalan and Leonard Schulman, "A coding theorem for distributed computation," in *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*. 1994, STOC '94, ACM.
- [13] Ravi Teja Sukhvasi and Babak Hassibi, "Error correcting codes for distributed control," *CoRR*, vol. abs/1112.4236, 2011.
- [14] Jing Zhou and Qian Wang, "Convergence speed in distributed consensus over dynamically switching random networks," *Automatica*, vol. 45, no. 6, 2009.
- [15] LJ Schulman, "Coding for interactive communication," *Information Theory, IEEE Transactions on*, vol. 42, no. 6, pp. 1745 – 1756, 1996.